LA PREPARATION DE DOCUMENTS

L'environnement Unix offre un ensemble d'outils pour aider dans la préparation de documents, tels que des lettres, des rapports, des livres, etc... Ces outils contiennent des programmes de formatage de texte permettant de spécifier la taille d'une page, l'espacement, les bords et les jeux de caractères à utiliser. La combinaison avec d'autres outils permet la définition de tables, de dessins ou d'équations mathématiques.

Ces outils, liés à un éditeur de texte facilitent l'écriture, la correction et la mise en page de documents. Il existe également des outils de correction d'erreur de frappe et de construction d'indice.

Nroff Et Troff

Les outils de base sont les formateurs de texte troff (prononcé : ti-roff), et nroff (prononcé : enne-roff). nroff est utilisé pour avoir des sorties de qualité terminal ou machine à écrire. troff est utilisé pour des imprimantes laser ou des photo-composeuses. En général, les fichiers troff sont compatibles avec nroff : les commandes supplémentaires de troff sont, en général, ignorées par nroff.

Le matériel sousmis à troff (ou nroff) contient à la fois du texte et des instructions indiquant comment le texte doit être imprimé. Les instructions troff commencent par un caractère spécial, en général les caractères ', . ou le caractère \. Même s'il n'y a aucune instruction (donc que du texte) troff formate le texte en utilisant quelques définitions primitives par défaut.

Avant d'entrer un document, vous devriez savoir comment vous voulez le formater, et quels packages de macros vous voulez utiliser. Il existe un ensemble de packages macro à l'usage général - le package mm est probablement le meilleur. Il est distribué avec le Système V de Unix. On devrait utiliser des macros pour les en-têtes, des têtes de chapitres, les bas de pages, des paragraphes et des 'displays'.

Lancement de troff

troff est activé comme :

troff [options ...] document

ce qui aura comme effet d'imprimer le document, commençant par défaut à la page no 1. Les options disponibles sont (entre autres) :

-m **p** utilise le package de macros p
-n **q** positionne la première page au numéro q
-o **list** imprime seulement les pages precisées dans list
-s **n** arrête tous les n pages
Par exemple,

troff -o3-5,8,12- chapitre1

imprime les pages 3 à 5, la page 8 et à partir de la page 12 le reste du fichier chapitre1.

Les instructions simples

Dans sa forme la plus élémentaire, troff lit des lignes d'entrée jusqu'à ce qu'il y ait suffisamment de caractères pour remplir une ligne de sortie. Si une ligne commence par un espace, toute sortie non encore imprimée sera imprimée et troff débute une nouvelle ligne. Troff débute une nouvelle ligne également au rencontre de lignes vides.

Les instructions troff apparaissent (seules) dans des lignes débutant par un . (un point) en première colonne. Par exemple,

.sp

est équivalent à une ligne vide dans l'entrée - c'est la commande pour un espacement vertical. Les instructions troff peuvent avoir des arguments, comme en :

.sp 5

qui est la commande pour 5 espacement verticaux, donc pour 5 lignes vides.

Les fonctions sont similaires aux commandes, mais elles peuvent apparaître au milieu du texte. Elles sont précédées d'un caractère \. Ainsi un commentaire troff s'écrit \" comme dans :

.sp 5 \" 5 lignes blanches

Ci-dessus une liste des commandes les plus usuelles :

.ce **n** centre les n lignes suivantes.

.ls **n** positionne l'interligne à n espaces verticaux. Par défaut, troff et nroff formatent avec un simple interligne.

.in **n** indente la marge gauche à la $n^{ième}$ position. L'argument peut être donné également en valeur relative, c'est-à-dire : en valeur relative à la position actuelle. Ainsi, la commande .in +4, incrémente la marge gauche de 4 positions. Notons également qu'on peut spécifier tous les arguments en mesure métrique : la commande .in +1i indente la marge gauche de un inch supplémentaire, et la commande .in +2.5c

indente la marge gauche de 2,5cm supplémentaire. Cette possibilité de mesure existe pour les arguments de tous les commandes ou cette possibilité a un sens.

.ti **n** — indente temporairement la marge gauche à la n[ième] position. Temporairement veut dire : que la ligne suivante est indentée comme spécifiée dans la commande, les lignes après gardent la marge spécifiée préalablement.

.ll **n** — positionne la longueur d'une ligne, donc la marge droite. n devrait être donné en inch ou centimetres. Initialement, la ligne a une longueur de 6.5 inch. Naturellement, n peut être donné en valeur relative, ainsi la commande .ll -2c réduit la longueur de la ligne de 2 centimetres - ou : agrandi la marge droite de 2 cm.

.fi — est la commande disant à troff d'ajuster le texte.

.nf — commande troff indiquant que le texte suivant ne doit pas être ajusté

.na — indique à troff de ne pas ajuster la marge droite. En fill-mode ceci implique que le texte est aligné à gauche, mais non à droite.

.ad — indique à troff que le texte doit être ajusté à gauche et à droite

.br — commande pour aller à la ligne

.pl — permet de changer la longueur d'une page. Par défaut elle est de 11 inch.

.po — ajuste la position de la sortie relative à la marge gauche physique du papier de l'imprimante.

Le tableau ci-dessus montre la relation entre po, in, ll et ce : la longueur de ligne est mesurée entre la marge gauche

et la marge droite; l'indentation est mesurée à partir de la marge gauche; le centrage se joue entre l'indentation actuelle et la marge droite; le page-offset po déplace la feuille troff sur le papier physique.

Revenant un instant sur la résolution des distances. Des mouvements peuvent être exprimés dans tout résolution dépendant du péripherique de sortie. Une imprimante à lignes n'a comme résolution (en général) que la ligne et le caractère. Par contre, une Diablo à une résolution verticale de 1/48ième inch. Le CAT typesetter a une résolution verticale de 1/144ième inch. L'unité interne de troff dépend donc du péripherique. Toutes les commandes qui spécifient une distance, peuvent être exprimées dans les unités définies ci-dessous :

nroff arrondi les distances horizontales ou verticales aux valeurs les plus proches permises par le péripherique de sortie.

Continuons la liste des commandes de base :

.ta $\mathbf{n_1\ n_2\ n_3}$ ...

positionne des tabulations aux distances indiquées. En général, les tabulations sont justifiées à gauche, mais l'on peut fait suivre l'indication de distance par la lettre R indiquant ainsi qu'on veut une justification à droite. Par exemple la commande :

```
.nf
.ta 2c 4c 7cR
```

suivi de :

(ou '→' représente le caractère de tabulation) sera formaté (approximativement) comme :

De même, pour centrer un champ, on peut utiliser le postfix C

.tc **c** spécifie le caractère c utilisé pour remplir les tabulation. En général, c'est un caractère blanc. Pour voir l'utilité :

```
.ta 10
nom→téléphone
.tc .
Pierre→45454545
```

donnera quelque chose comme :

```
nom            téléphone
Pierre................45454545
```

.bp est la commande troff pour commencer une nouvelle page.

.ne **n** continue la sortie s'il reste au moins n lignes sur la page. S'il restent moins, troff fait un bp

.tl 'arg_1'arg_2'arg_3'

produit un titre de page. arg_1 est justifié à gauche, arg_2 est centré, et arg_3 est justifié à droite. Par exemple, la commande

.tl 'Unix'Draft'Chapitre 5'

imprimera :

Unix	Draft	Chapitre 5

Les Macros

Une macro permet de grouper une suite de commandes ou de textes sous un nom et de l'activer à travers ce nom. Avant d'être utilisée, une macro doit être définie. Dans la définition d'une macro, on peut utiliser des définitions déjà existantes. Une macro est définie avec la commande .de. La définition se termine par la commande ...

A titre d'exemple, voici une petite macro définissant des débuts de paragraphes :

```
.de PP
.sp
.ti 4
.ne 2
..
```

demandant une ligne blanche et une indentation temporaire de 4 unités, et ceci seulement s'il reste au moins deux lignes sur la page. S'il en reste moins, un page break sera effectué. Si un paragraphe débute par un appel de cette macro, comme dans :

```
.PP
ca produit un ....
```

ca produit un nouveau paragraphe, dont la première ligne est indentée de quatre caractères. En général, on utilise des noms en caractères majuscules pour ne pas entrer en conflit avec des noms utilisés par troff. Les macros peuvent avoir des arguments, comme dans

```
.SH 5.1.3  "Les Macros"
```

les arguments sont séparés par des espaces et doivent être entourés par des guillemets s'ils contiennent un caractère espace ou tabulation. A l'intérieur de la définition, les arguments sont accédés (comme en shell) par \$1, \$2, etc, jusqu'à neuf arguments. Les arguments pour lesquels des valeurs n'ont pas été livrées à l'appel de la macro sont initialisés à la chaîne vide. La macro SH aurait pu être définie comme suit :

```
.de SH    \" l'en-tête d'une section
.sp
\&\\$1 \\$2
..
```

Une définition de macro est lue deux fois : une première fois pendant la définition, et une deuxième fois à l'appel. Dans les deux cas, troff évalue les arguments. Dans la définition ci-dessus, le caractère \ est nécessaire avant \$1 pour éviter que troff interprète \$1 pendant la définition. En général, le caractère \ quote son argument (comme dans le shell).

L'utilisation des commandes introduites jusque là, permet de formater relativement aisément du texte ordinaire. Néanmoins, le problème du contrôle de formatage d'une page devient vite apparent. Il n'existe pas une commande simple contrôlant le haut et le bas de page. Par contre, on peut positionner des trappes à des positions verticales à l'intérieur d'une page. Quand cette position verticale est atteinte, la commande spécifiée sera executée. Souvent, on pose deux trappes : l'une pour le haut de la page et l'autre pour le bas de la page. Le haut de page se pose comme :

```
.wh 0 ...
```

et l'autre près du fin de la page comme, par exemple,

```
.wh -5 PE
```

Ce dernier pose une trappe à 5 lignes de la fin de la page. Après avoir sorti la ligne 5 lignes avant la fin de la page, la macro PE est activée. La définition de cette macro est séparée de la spécification de la trappe. Par exemple, la macro suivante :

```
.de PE
‘sp 2
.tl :Unix Manual:First Edition 1971:Page %:
‘bp
..
```

définit PE de manière à faire un saut de deux lignes et d'imprimer un titre. Le séparateur : utilisé ici dans la commande tl est tout à fait légal : tl prend le premier caractère non blanc en tant que séparateur de ces trois champs. Nous l'avons utilisé ici, puisque le séparateur standard (‘) fait partie de la chaîne imprimée.

Des commandes débutant par un caractère quote (‘), plustôt que par le caractère point, évitent que des lignes (ou des parties de lignes) déjà traitées soit imprimées. Elles sont alors gardées jusqu'à la fin du traitement de la macro; ici elles ne s'imprimerons pas avant la page suivante.

Le caractère spécial %, quand il est utilisé à l'intérieur d'un titre (dans une commande tl), imprime le numéro de page courant en decimal.

Déplacements horizontaux / verticaux

Des déplacements verticaux à l'intérieur d'une ligne peuvent être obtenus avec les fonctions \d, \u ou \v. \d continue la sortie une demie ligne plus bas (c'est pour des indices), \u continue la sortie une demie ligne plus haut (c'est pour les superscripts), et \v prend en argument une distance verticale arbitraire, indiquant la position à, laquelle l'impression continuera. v'1c' déplace d'un centimetre vers le bas, et v'-1c' continue l'impression un centimetre plus haut. Bien entendu, des mouvements verticaux à l'intérieur d'une ligne devraient s'annuler mutuellement, si l'on ne veut pas avoir des résultats surprenants.

Des mouvements horizontaux peuvent être spécifies de manière similaire avec la fonction \h, dont l'argument doit être la distance à aller vers la droite, si l'argument est positif, à aller vers la gauche, si l'argument est négatif. Calculer la distance de déplacement horizontale peut prendre encompte la longueur d'une chaîne. C'est pourquoi la fonction \w existe. Elle permet de calculer la longueur d'une chaîne. Ainsi,

\w'ceci'

livre la longueur de la chaîne, ceci en unités de base. Par exemple :

ceci\h'-\w'ceci'u'ceci

reécrit le mot ceci, le faisant ainsi apparaître plus foncé (en bold). Le même effet peut être obtenu en marquant la position horizontale avec la fonction \k dans un registre et de se déplacer plus tard vers cette position.

\kxceci\h'|\nxu'ceci

a exactement le même effet. la fonction \kx marque la position horizontale et la stocke dans x, la fonction \h'|...' se déplace vers la position horizontale spécifiée, et la fonction \nx donne la valeur du registre x.

La place verticale peut également être marquée; c'est la commande mk. La commande rt retourne vers cette position verticale à l'intérieure de la page. Ceci permet la sortie en plusieurs colonne. Par exemple, la Figure sur la page suvante montre comment imprimer deux colonnes, la première avec l'indentation actuelle et la deuxième indentée d'un centimètre supplémentaire.
ce qui donne le tableau suivant :

On peut dessiner des lignes horizontales et verticales. La fonction \l'1c' trace une ligne horizontale de 1cm (comme ceci : _____). On peut spécifier le caractère utilisé pour tracer la ligne, ainsi \l'1c.' trace une ligne de un centimetre en pointillée (comme ceci : ....). La fonction \L est similaire pour des lignes verticales. Par exemple :

\kx\l'1i'\L'2'\l'-1i'\h'|\nx'\L'-2'

dessine une boîte d'une longueur de un inch et d'une hauteur de 2 lignes.

Les conditionnelles

Les commandes peuvent n'être activées que suivant certaines conditions. Les conditions peuvent être des comparaisons de chaînes ou des expressions arithmétique. La forme la plus simple est :

.if c qqc

où, si la condition c est vraie, qqc est utilisée en tant qu'entrée. qqc peut être du text ou des commandes troff. Les conditions ont été décrite dans le tableau ci-dessus (o, e, t et n). Par exemple :

```
.if e .tl '%'''
.if o .tl '''%'
```

place le numéro de page sur la gauche des pages à numéros paires et sur la droite des pages à numéros impaires. D'autres conditionnelles sont :

.if N qqc

qui accepte qqc si l'expression numérique N est supérieure à zéro.

.if 'chaîne$_1$'chaîne$_2$' qqc

accepte qqc si chaîne$_1$ est identique à chaîne$_2$. L'opérateur ! peut être utilisé pour inverser le sens du test :

.if !N qqc

accepte qqc si l'expression N est inférieure ou égale à zéro. Il existe également une forme if-then-else :

.ie c qqc$_1$
.el qqc$_2$

Si c est vraie alors qqc$_1$ est acceptée, sinon ce sera qqc$_2$. Si plusieurs actions doivent être spécifiées, elles doivent être entourées par les délimitateurs \{ et \} comme ci-dessous :

Les registres de chaînes

Si l'on doit souvent utiliser une chaîne de caractères, on a intérêt de la définir dans un registre de chaînes. Les registres et les macros partagent l'espace de noms : attention à ne pas accidentellement détruire une macro! Pour définir un registre de chaîne on écrit :

.ds xx la-chaîne

où chaîne commence avec le premier caractère non blanc ou après un guillemet (") initiale. Si, par exemple, on a souvent besoin de la boîte définie plus haut : \kx\l'1i'\L'2'\l'-1i'\h'|\nx'\L'-2', on peut la définir ainsi :

```
.ds b1 \kx\l'1i'\L'2'\l'-1i'\h'|\nx'\L'-2'
```

Cette chaîne pourra être utilisée ensuite dans le texte comme :

*(b1

et à cet endroit apparaîtra alors la boîte définie. Naturellement, on peut également définir des registres à un nom avec un seul caractère. Par exemple

.ds v “Université Paris VII - Vincennes à Saint-Denis

peut être accedée simplement par *v. Notez les guillemets initiales, ils sont nécessaires puisque la chaîne contient des caractères blancs.

Les registres de nombres

troff permet également de définir des registres de nombres pour y stocker des valeurs numériques. Ces registres peuvent être initialisés, incrémentés et decrémentés, ainsi que - naturellement - consultés. Les noms de ces registres se composent également de un ou deux caractères et leurs valeurs peuvent être accédés soit comme \nx ou \n(xx. Heureuse nouvelle : les noms des registres de nombres ne sont pas en conflit avec les noms des macros et des registres de chaînes. Ces registres peuvent être utilisés dans des expressions arithmétiques. En troff, par exemple, le jour, le mois et l’année depuis 1900, sont gardés dans les registres numériques dy, mo et yr de manière telle que

19\n(yr \n(mo \n(dy

imprimerait

1988 5 1

si la date est le premier May 1988. La conversion par défaut des nombres est dans une représentation romaine, comme dans l’exemple ci-dessus. Mais la commande af permet de changer le format :

.af R f

R est un registre et f est un des formats ci-dessous :

troff ne définit pas une chaîne pour le nom du mois, mais on peut le faire comme suit :

```
.if \n(mo-0 .ds MO Janvier
.if \n(mo-1 .ds MO Fevrier
...
```

Les registres de nombres sont initialisés avec la commande nr, par exemple :

.nr IN 2

ou

.nr IN +2

ajoute 2 à la valeur de IN. Les registres peuvent être incrementés ou decrémentés par :

\n+x ajoute m à x
\n+(xx ajoute m à xx
\n-x soustrait m à x
\n-(xx soustrait m à xx

La grandeur de **m**, ajoutée ou soustraite est positionnée par la commande nr également :

.nr sn 0 1

met le registre sn à 0 et positionne son incrément à 1. Il y a d'autres commandes qui affectent des registres :

.mk px

par exemple, met dans le registre px la position verticale actuelle (en unité de base). Cette valeur peut être utilisée ensuite pour retourner à cette place. La commande ".sp |\n(pxu" peut servir à cette tâche. Notez la barre verticale en tant que directive de positionnement absolue. Les registres du tableau suivant sont positionnés par troff mais ne peuvent pas être modifiés par l'utilisateur; néanmoins, ils peuvent être consultés.

Les registres du tableau de la page suivante sont positionnés par troff. Ils peuvent être consultés et modifiés par l'utilisateur.

Les jeux de caractères

nroff ne peut distinguer qu'entre trois jeux de caractères : les caractères normaux (rien à dire sur cela), les caractères bold qui sont imprimés en surimpression et les caractères italiques, qui seront imprimés en soulignés. Le changement de ces fonts se fait par la commande ft :

.ft B pour changer vers le jeux de caractères bold
.ft R pour changer vers le jeux de caractères normal
.ft I pour changer vers le jeux de caractères italiques

Pour changer à l'intérieur d'une ligne, la fonction \f peut être utilisée: respectivement \fB pour changer vers bold, \fR pour changer vers normal et \fI pour changer vers les caractères italiques. Pour revenir à un instant donné vers le jeux de caractères précédent, la fonction \fP ou la commande

.ft P

peut être utilisée.

En troff il est possible de changer (réellement) le jeux de caractères. Par défaut troff utilise les jeux de caractères standard Times Roman, Times bold et Times italique. Les jeux de caractères disponibles dépendent bien entendu du périphérique uilisé. Ainsi, la fonction \f(HItest\fP imprimerait test en Helvetica italique et retournerait ensuite au jeux de caractères précédent.

Troff permet également de changer la taille des caractères avec la commande ps (pour point-size) ou avec la fonction \s. Par exemple, la macro :

```
.de SH
.ps +2
\&\\$1 \\$2
.ps -2
..
```

redéfinit celle que nous avons préalablement définie en changeant temporairement la taille des caractères. Pour changer l'espacement entre les lignes, la commande vs fait l'affaire. (En général, l'espacement vertical est de deux points supérieurs à la taille des caractères.

Autres Outils Pour Le Traîtement De Documents

Vous l'avez bien vu : programmer en troff/nroff est comme programmer en assembleur. Troff est vraiment un langage de très bas niveau. C'est pourquoi on trouve des package de macros standard. Les packages les plus connus sont :

ms	un package simple pour formater du texte ordinaire
mm	un autre package pour formater du texte
tbl	un package pour formater des tableaux
eqn	un package pour formater des expressions mathématiques

A ces packages s'ajoutent le programme col, un postprocesseur qui filtre des reverse line-feeds produit par nroff afin de pouvoir l'imprimer sur des imprimantes non sophistiquées, le programme spell qui, comme son nom l'indique, verifie l'orthographe de vos textes et le programme style qui analyse (de manière simple) la structure de vos phrases (ceci n'existe que pour du texte écrit en anglais : les résultats obtenus en donnant du texte français à style sont parfois très surprenants !).

Le package mm

La tâche du macro-package mm est principalement de vous livrer les combinaisons de commandes troff les plus utilisées. Ainsi, mm offre une macro numérotant automatiquement les chapitres et sous-chapitres, une autre prenant en charge l'indentation et, si nécessaire, la numérotation de listes, ou encore une pour inserer vos notes de bas de pages au bon endroit. Ci-dessous, la liste des commandes mm les plus usuelles. Notez que, si vous utilisez ce package, vous devez l'annoncer pendant le lancement de troff comme :

troff -mm vos-fichier-a-troffer

pour inclure le packet mm ou

troff -ms vos-fichier-a-troffer

pour inclure le packet ms. Voici alors la liste de commandes mm :

.AS debut du résumé, se termine par .AE.

.AU l'argument est le nom de l'auteur

.DF permet de garder une suite de textes ensembles. Si la totalité du texte prend plus que l'espace encore disponible sur le reste de la page, tout le texte sera alors imprimé sur la page suivante. C'est trés utile pour la mise en page de tableaux, par exemple. DF doit se terminer par la commande .DE.

.EQ le texte se trouvant entre cette commande et la commande .EN doit correspondre à des comman-des pour eqn, le package pour la mise en page d'expressions mathématiques.

.FS indique le début d'une note de bas de page. La note doit se terminer par .FE.

.H n "..."

l'entête de chapitre de niveau n. Les chapitres sont, comme dans ces notes, organisés en arborescence : le chapitre 1, contient, par exemple, les sous-chapitres 1.1 et 1.2. Le chapitre 1 est de niveau 1, les sous-chapitres sont de niveau 2, les sous sous chapitres de niveau 3, ainsi de suite jusqu'au niveau 6. mm procède automatiquement à la numérotation, de manière telle que si l'on introduit entre le sous chapitre 1.1 et 1.2 un nouveau sous chapitre, avec la commande

.H 2 "un titre"

ce nouveau sous chapitre sera automatiquement numéroté 1.2, et l'ancien sous chapitre 1.2 deviendra le sous chapitre 1.3.

.TS début d'une table. Le texte entre TS et la prochaine commande .TE sera consideré comme entrée pour le package de macro tbl.

Nous arrêtons ici la description des commandes mm. Le lecteur intéressé pourra se référer au manuel d'utilisation.

Le package tbl

tbl accepte une description simple d'un tableau et génère des commandes troff pour l'imprimer. Les données pour tbl sont du texte mélangé avec des informations sur le formatage du tableau. Les commandes tbl doivent être entre la commande .TS et .TE, comme en :

```
.TS
la description du tableau
.TE
```

tbl laisse tout autre texte inchangé. Les commandes TS et TE sont également laissées dans l'état et peuvent donc être utilisées par d'autres macros. Les tableaux sont décrits comme suit :

```
.TS
option de layout du tableau;
format des colonnes et lignes.
données du tableau
.TE
```

Les signes ; et . sont à prendre littéralement et doivent terminer les sections appropriées. Dans l'exemple suivant, → designe le caractère de tabulation qui doit séparer des colonnes.

```
.TS
center;
c  s s
c c c
l l n.
Les Candidats
Nom→Prénom→Téléphone
Chirac→Jacques→45454545
Mitterand→François→47777777
Giscard d'Estaing→Valérie→44444444
d'Arc→Jeanne→499900990
.TE
```

Cet exemple produit le tableau suivant :

Dans la section des instructions de formatage de tbl, les options suivantes sont disponibles :

c une entrée de colonne centrée.

l une entrée de colonne justifiée à gauche

n une quantité numérique, alignée sur les unités

r une entrée de colonne justifiée à droite

s un entête élargi, i.e. l'entrée de la colonne précédente continue dans cette colonne. Ainsi, dans notre exemple, la première entrée, Les Candidats, est centrée sur la totalité du tableau.

L'option center à comme effet de centrer sur la page la totalité du tableau. Par défaut, les tableaux sont justifiés à droite de la page. Les autres options disponibles sont :

allbox un cadre est dessiné autour de chaque entrée du tableau
box un cadre est dessiné autour du tableau.
center le tableau est centré sur la page

doublebox
un cadre en double ligne est dessiné autour du tableau

tab(**x**) x est utilisé pour séparer des colonnes

Le package eqn

Le préprocesseur eqn est utilisé pour la mise en page d'équations mathématiques pour des imprimantes de haute résolution. Le package neqn, acceptant les mêmes commandes, met en page des expression mathématiques sur des imprimantes d'une résolution similaire aux machines à écrire. Comme pour tbl, les commandes eqn (ou neqn) sont entourées par deux commandes particulières : .EQ et .EN. Prenons un exemple simple :

```
.EQ
x=a+b
.EN
```

produit la version en caractères italiques ci-dessous :

$$x=a+b$$

Le text de eqn peut également être inclus entre des délimitateurs définis par l'utilisateur :

```
.EQ
delim $$
.EN
```

par exemple, définit le caractère "$" en tant que délimitateur de eqn. Ainsi

Soit $ 2 pi omega $ le ...

se traduit en :

Soit $2\pi\omega$ le

eqn est un préprocesseur qui produit des instructions pour troff ou nroff. Pour n'en donner que quelques exemples – le lecteur se référera au manuel pour de plus amples renseignements – ci-dessous 3 commandeσ eqn et le résultat qu'elles produisent :

```
.EQ
G(z)~mark =~ e sup {ln ~ G(z)}
~=~ exp left (
    ~sum from {k >= 1}
      {~~S sub k z sup k} over k right )
~=~ prod from {k >= 1}
     ~~e sup {S sub k z sup k / k}
.EN
```

produit le résultat suivant :

$$G(z) = e^{\ln G(z)} = \exp\left(\sum_{k \geq 1} \frac{S_k z^k}{k} \right) = \prod_{k \geq 1} e^{S_k z^k / k}$$

le programme ci-dessous :

```
.EQ
bold V bar sub nu ~=~
left [ lpile { a above b above c } right ] +
left [ matrix { ccol { A(11)~~ above . above . }
                ccol { . above . above . }
               ccol { . above . above ~~A(33) } }
right ] cdot
left [ pile { alpha above beta above gamma }
right ]
.EN
```

produit l'equation :

$$\bar{\mathbf{V}}_{\mathbf{n}} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} + \begin{bmatrix} A(11) & . & . \\ . & . & . \\ . & . & A(33) \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$

et finalement :

```
.EQ
a sub 0 + b sub 1 over
   { a sub 1 + b sub 2 over
      { a sub 2 + b sub 3 over
         { a sub 3 + b sub 4 over
            { a sub 4 + ... }}}}
.EN
```

qui produit en sortie :

$$a_0 + \frac{b_1}{a_1 + \frac{b_2}{a_2 + \frac{b_3}{a_3 + \frac{b_4}{a_4 + \ldots}}}}$$

Notons finalement que, puisque tbl et eqn sont des préprocesseurs, si vous utilisez à la fois tbl et eqn, il est conseillé d'activer d'abord tbl et ensuite eqn, comme dans :

tbl **fichier** | eqn | troff -mm

Spell et Style

Les commandes spell et style n'ont pas trop d'intérêt pour l'utilisateur français quand il écrit dans sa langue natale. Mais quelle utilité si vous devez écrire un papier en anglais ! spell compare les mots qu'il reçoit avec ceux contenus dans un dictionnaire en ligne, et sort une liste de mots non contenus dans ce dictionnaire. Cette liste peut contenir des mots correctement écrits mais non contenus dans le dictionnaire. style fait une analyse de surface d'un document et rapporte sur la lisibilité, longueur de phrases, structures de phrases, longueur de mots, types de verbes, usage de mots, et sur des méthodes pour commencer les phrases. Principalement, cet outil livre des indications utiles si l'on surutilise certains mots ou certaines structures grammaticales.

Pour terminer ce chapitre sur le traitement de texte et le formatage de documents, nous donnons ci-dessous un programme nroff sousmis à votre étude. Ce merveilleux programme, écrit par Larry Wall de System Development Corporation à Santa Monica, est l'unique exemplaire (à ma connaissance) d'un programme nroff autoréproducteur. Le voici :

.ec8
.emZZ
.nryy1000-12
.nrzz100-1
.nf
.deXX
.tr88n(yy
88!88!88$1
.tr88n(zz
.amZZ..
88!88!.XX88$1
.nrnl0-1
...
..
.XX.ec9
.XX.emZZ
.XX.nryy1000-12
.XX.nrzz100-1
.XX.nf
.XX.deXX
.XX.tr99n(yy
.XX99!99!99$1
.XX.tr99n(zz
.XX.amZZ..
.XX99!99!.XX99$1
.XX.nrnl0-1
.XX...
.XX..